

REMARKS/ARGUMENTS

Applicant graciously appreciates the Office's attention to the instant application. A Request for Continued Examination (RCE) under 37 CFR §1.114 is filed herewith.

A telephonic conference was held on November 16, 2005 where Applicant's representative and the Examiner discussed "just-in-time" (JIT) features of various subject matter, for example, as in claims 5, 6 and 18. Applicant further notes that the instant specification refers to "lazy loading", in particular, "conversion with lazy loading" (see, e.g., method 600 of Fig. 3). Applicant suggested an amendment in an effort to clarify that various conversions methods operate on a "lazy" or just-in-time (JIT) basis.

In view of the telephonic discussion and the following remarks, Applicant respectfully requests reconsideration and allowance of the pending claims of the instant application. This response is believed to be fully responsive to all issues raised in the September 8, 2005 Office Action. Claims 1, 5-8, 12, 15, 19, 20 and 24 are currently amended, claims 10 and 18 are canceled; thus, claims 1-3, 5-9, 11-12, 15-17, 19-24 are pending.

Rejection of Claim 15 under 35 U.S.C. §112¶2

Applicant currently amends claim 15 to recite "intermediate language code framework" to clarify the relationship of claim 15 to claim 12. Applicant submits that claim 15 now satisfies the requirements of §112¶2.

Rejection of Claims 1-3, 5-12, 15-24 under 35 U.S.C. §103

In the Office Action mailed September 8, 2005, the Office rejected claims 1-3, 5-12 and 15-24 as being obvious over U.S. Patent No. 6,553,405, to Desrochers in view of U.S. Publication No. 2002/00736236 to Helgeson and further in view of Admitted Prior Arts (APA).

As mentioned with respect to the teleconference of November 16, 2005, Applicant suggested amendments to more particular bring forth aspects of “lazy” or just-in-time (JIT) operation. Applicant respectfully directs the Office to the exemplary method 600 of Fig. 3 “Conversion with Lazy Loading”. While loading is “lazy” in the method 600, so is conversion of the lazily loaded code for the reference class (block 630) as it depends on the loading (block 626). Amendments presented above and discussed below aim to more particularly claim conversion of code (e.g., from bytecode framework to intermediate language code framework) lazy conversion of reference class code (if required) and execution of reference class code (or converted reference class code). In the method 600, the reference class stems from a reference in code that is associated with a framework (e.g., bytecode) that differs from the type of framework (e.g., intermediate language code) on which execution takes place. Thus, a need for conversion arises for at least some of the code.

To more particularly claim various subject matter of the instant application, Applicant currently amends:

independent claim 1 to include “*executing code for the referenced class on the intermediate language code framework*”;

independent claims 8, 19 and 24 to include “*executing the converted code for the referenced class on the intermediate language code framework*”;

independent claim 7 to include “*to initiate execution of the converted reference class code on the intermediate language code framework*”; and

independent claim 12 to include “*the intermediate language code framework executes the converted class file code*”.

Thus, all of the independent claims now recite language germane to execution of code for a reference class where the reference class was referenced in a code associated with a bytecode framework. Such execution of reference class code may be said to occur in a lazy manner. Further, the reference class code may be converted or other at the time of execution. Claims 8, 19 and 24 recite

1 “converted code for the referenced class”, claim 7 recites “converted reference
2 class code” and claim 12 recites “converted class file code”. Hence some of the
3 claims explicitly recite conversion for reference class code.

4 While claim 1 does not recite conversion of the reference class code, such
5 subject matter is included in dependent claim 5, which recites “*wherein the code*
6 *for the reference class comprises code associated with a bytecode framework and*
7 *further comprising converting the code for the referenced class to a converted*
8 *reference class code capable of execution on the intermediate language code*
9 *framework*”. Thus, claim 5, as it depends on claim 1, recites a lazy conversion of
10 code for a referenced class.

11 Claims 2, 3, 5 and 6 depend on claim 1, claims 9 and 11 depend on claim 8,
12 claims 15-17 depend on claim 12 and claims 20-23 depend on claim 19.

13 Applicant submits that these dependent claims are patentable over the references
14 of record for at least the same reasons as the independent claims.

15 Response to Office's Proffered Evidence

16 With respect to the evidence relied on by the Office in rejection of the
17 claims (Desrochers, Helgeson and APA), Applicant respectfully offers
18 clarification.

19 Desrochers Reference

20 The Office states that the Desrochers reference teaches “converting”;
21 however, this “converting” is not the type of converting recited in the claims. The
22 “converting” of the Desrochers reference is the customary process used in
23 bytecode frameworks such as the JAVATM framework (again, the Desrochers
24 reference discloses and teaches use of the JAVATM framework only, see, e.g., col.
25 3, lines 36-44). The “converting” of the Desrochers reference does not intend in
any way to transform code associated with a bytecode framework to code capable
of execution on an intermediate language framework. Instead, it is a type of

1 “conversion” associated with the intrinsic operation of a single type of framework,
2 i.e., a bytecode framework like the JAVA™ framework. Applicant further finds
3 that the Derochers reference fails to provide evidence sufficient to suggest any
4 need for another, different type of framework.

5 *Helgeson Reference*

6 With respect to the Helgeson reference, Applicant fails to find evidence
7 sufficient to suggest a framework other than the JAVA™ framework. For
8 example, the Helgeson reference mentions only one type of virtual machine, the
9 JAVA™ virtual machine (see the Helgeson reference at ¶¶215, 379, 523, and
10 1201). Further, at paragraph 423, the Helgeson reference states:

11 In the alternative embodiment, every class in an application does not have to be a
12 bean. Indeed, with the overhead of locating a bean through a naming service and
13 going through the home and remote interfaces of a bean to perform useful work
14 would negatively impact performance (though some servers will optimize the
15 process for beans located within the same virtual machine). The application
16 developers can implement selected classes as helper classes and not as beans. Sun
17 Microsystems' J2EE Application Programming Model identifies certain instances
18 where helper classes are applicable. One such example is dependent classes that
19 can only be accessed indirectly through other classes (beans). Sun's J2EE APM
20 offers CreditCard and Address classes as examples of a dependent classes.

21 Thus, in the instance where every class need not be a bean (i.e., a JAVA™
22 bean), it is nevertheless a class associated with the JAVA™ framework (Sun's
23 J2EE APM). In contrast, various claims recite conversion of a class associated
24 with a bytecode framework to a class capable of execution on a different type of
25 framework, e.g., an intermediate language code framework. Such subject matter is
not taught or suggested by the Desrochers reference or the Helgeson reference,
alone or in combination.

23 *APA*

24 The Office also points to Admitted Prior Arts (APA) for the fact that
25 different types of frameworks existed at the time of filing of the instant

1 application. However, such evidence is insufficient to overcome that which is
2 lacking in the Desrochers reference and the Helgeson reference. Again, the
3 combination of Derochers and Helgeson fails to teach or suggest conversion of
4 code for different types of frameworks.

5 **Conclusion**

6 Applicant submits that pending claims 1-3, 5-9, 11-12, 15-17, 19-24 are in
7 condition for allowance. Applicant respectfully requests reconsideration and prompt
8 issuance of the subject application. If any issues remain that prevent issuance of this
9 application, the Office is urged to contact the undersigned attorney before issuing a
10 subsequent Action.

11
12
13 Respectfully Submitted,

14
15 Dated: 12/8/05

16 By: 

17 Brian Pangle
18 Reg. No. 42,973
19 Lee & Hayes, PLLC
20 (509) 324-9256
21
22
23
24
25